

MAKING VIRTUAL BABY ALIVE

D. Korošec¹, L. P. Halamek², D. Zazula¹

¹Faculty of Electrical Engineering and Computer Science, University of Maribor, Slovenia

²Department of Pediatrics, Stanford University School of Medicine, CA, USA

Abstract - We have designed and implemented a prototype virtual environment for medical training in neonatal resuscitation. The central element of this environment is the dynamic virtual model (avatar) of a new-born child, built using VRML and Java. Physiological variables relevant for training were chosen to be represented through the avatar: heart rate, respiration rate, skin colour and activity level (such as movement and crying). Implementation mechanisms of presenting these vital signs on the virtual baby with animation and sound are described in this paper.

We are using time-dependent description of basic events to specify scenarios according to which the condition of a baby is changing. Sets of scenarios are then coupled to actions performed by a student during training. Such approach for all relevant variables had seemed good at first, but it soon became impractical and hard to follow due to separate treatment of variables. Therefore an attempt to create a simplified, easier controllable model with interdependent variables is being under development. This paper reports on current implementation as well as on the explored ideas for the future.

Keywords - Virtual environments, VRML, collaborative medical training, avatars, computer-aided education

I. INTRODUCTION

In our Virtual Delivery Room project (VIDERO)[1] we are developing a multi-user virtual environment for training of medical students in neonatal resuscitation. Its motivation and background are presented in Section II.

Sections III and IV are devoted to the virtual baby - the represented variables are listed, how their trajectories are described in scenarios with discrete events, and how we model the relations of these variables to the overall condition of the baby is explained.

In the fifth Section we then briefly present the program, which a medical specialist can use to tune and validate the implemented behaviour of a virtual baby.

II. SIMULATION TRAINING IN MEDICINE

Simulation training for health care professionals can be provided by simulated exercises. Good example of such training is the system of crisis management training in neonatal resuscitation[2] introduced by L. Halamek, D. M. Gaba and their team at the Veterans Affairs Palo Alto Health Care System[3]. It is necessary to face medical students with simulated situations, because some serious conditions are potentially very rare to encounter during practical placements of medical students. So-called 'learning by accident' phenomena can be avoided using medical simulators.

'Real-world' simulators can provide a significant degree of immersion and a very realistic experience, but the price is high: special training facilities with instruments, monitors, mannequins, and skilled senior staff to control and supervise the training are necessary. With the development of virtual reality (VR) technologies a strong trend is present toward implementing such simulators completely or partly as virtual environments. In fact, we believe that due to all the complexity of medical knowledge today a complete four stage learning path should be adopted as shown in Fig.1.

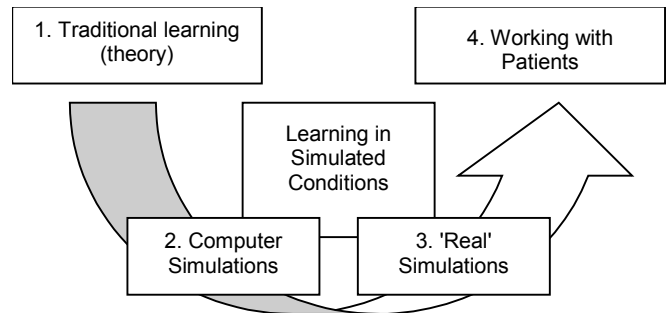


Fig. 1: Learning medicine today should be a step-by-step process.

For computer simulations standalone programs with built-in scenarios and predefined training tasks can be used, although only training involving a mentor can usually provide higher complexity training and realistic assessment.

With this in mind we designed our virtual environment as a multi-user collaborative training tool (Fig. 2), where a trainee and a mentor (as well as possible observers) share the same virtual space.

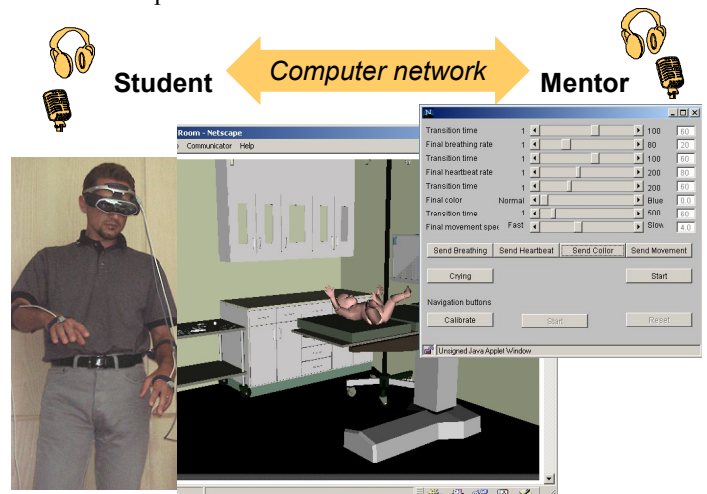


Fig. 2: Training in the multi-user virtual environment.

Report Documentation Page

Report Date 25OCT2001	Report Type N/A	Dates Covered (from... to) -
Title and Subtitle Making Virtual Baby Alive		Contract Number
		Grant Number
		Program Element Number
Author(s)		Project Number
		Task Number
		Work Unit Number
Performing Organization Name(s) and Address(es) Faculty of Electrical Engineering and Computer Science, University of Maribor, Slovenia		Performing Organization Report Number
Sponsoring/Monitoring Agency Name(s) and Address(es) US Army Research, Development & Standardization Group (UK) PSC 802 Box 15 FPO AE 09499-1500		Sponsor/Monitor's Acronym(s)
		Sponsor/Monitor's Report Number(s)
Distribution/Availability Statement Approved for public release, distribution unlimited		
Supplementary Notes Papers from the 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, October 25-28, 2001, held in Istanbul, Turkey. See also ADM001351 for entire conference on cd-rom., The original document contains color images.		
Abstract		
Subject Terms		
Report Classification unclassified	Classification of this page unclassified	
Classification of Abstract unclassified	Limitation of Abstract UU	
Number of Pages 4		

III. REPRESENTATION AND CONTROL OF A VIRTUAL BABY

Relevant and Observable State Variables (OSVs)

Together with medical doctors and according to the official training guidelines[1] we first identified all *relevant parameters*, which should be included in the model of a virtual new-born. Out of this variables, listed in the left column of Table I, we then compiled vital signs, which can be appropriately rendered: beating of the heart, respiration, skin colour and activity as movement and sounds (sighs, crying). Each of these signs is linked to its own so-called *observable state variable (OSV)*, each having its range of values: heart rate (HR) from 0-200 beats/min; respiration rate (RR) from 0-80 breaths/min; colour (Col) and activity (Mov) are normalised to [0,1] between both extreme levels.

TABLE I
RELEVANT AND OBSERVABLE STATE VARIABLES OF THE VIRTUAL BABY

Relevant State Variables	OSV	Observability / Representation
Weight, Size	-	-
Heart Rate	HR	sound (stethoscope), pulse reading, bedside monitor
Breathing Rate	RR	chest movement, sound, bedside monitor
O ₂ concentration	Col	pink, blue (central or peripheral cyanosis)
Blood Pressure	-	-
Condition of the lung vessels	-	-
Foetal lung fluid level	-	-
Blood pH	-	-
-	Mov	movement, sounds (crying)

Manual Control over OSVs - Mentor's Control Panel

In the current version of our training system it is the sole responsibility of the mentor (training supervisor) to control the values of all observable state variables. This can be done using a separate Java dialog (not seen by a trainee) shown in Fig. 3. The main controls there are sliders for setting the values of OSVs. Transitions from the current to newly set values will usually not be instant and abrupt in order to achieve realistic impression, therefore the *transition time* should be set with every change. During this time (500 seconds max.) smooth, linear transition from the current to the specified final value will occur. Parameter value starts to change after clicking the corresponding *Send...* button. This is registered in the system as an *event*.

It would be, of course, very difficult for the mentor to control and manually set every variable during the training. To overcome this difficulty we have implemented support for the predefined lists of events, called *scenarios*.

Event Scenarios

Scenarios are plain text files in which each line describes one event in the system. Events represent changes of the OSV values, format and some examples are in Table II. Mentor can load and start any of the predefined scenarios, scenarios can also be loaded automatically as responses to student actions.

TABLE II

EXAMPLE OF THE EVENT SCENARIO

Time [in sec]	Parameter ID	Final parameter value or percent of change	Time to reach the final value [in sec]
1	RR	12	1
1	Col	0.2	1
10	HR	150	30
15	Col	-10%	10
60	Mov	+20%	20

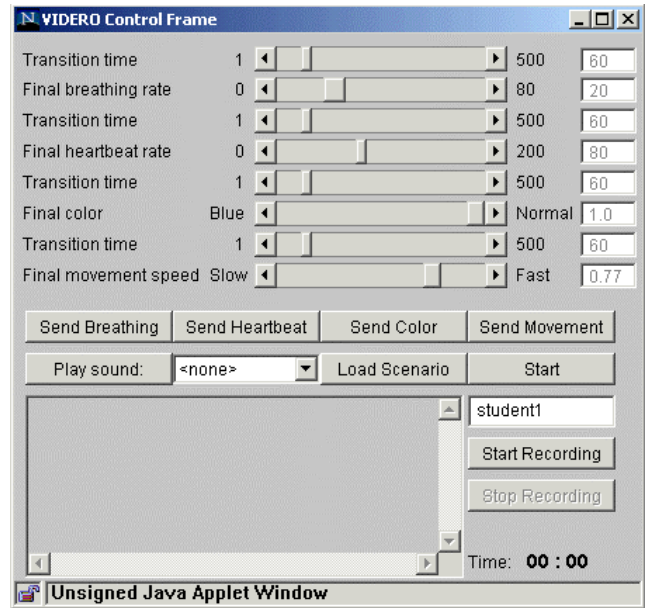


Fig. 3: Mentor's control panel for manual setting of OSVs.

Audio-Visual Representation of the Virtual New-born Baby

Graphic and auditory representation of our virtual baby is implemented in VRML [4]. 3D description of the baby avatar is stored in nodes of a scene graph according to *H-ANIM recommendation* [5]. Values of OSV are sent from Java application via the *External Authoring Interface (EAI)*. Most of the dynamics needed for representing movements, changing of colour and sounds is built using Java scripts and VRML 97 *interpolators* and *message routing mechanism* (Fig. 4).

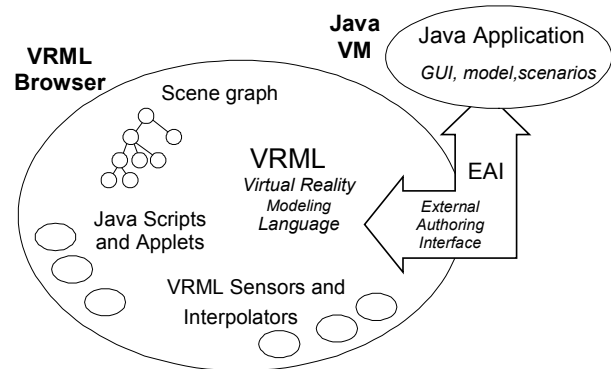


Fig. 4: Used programming technology

IV. LINKING VARIABLES FOR EASIER MANUAL CONTROL

Even with predefined lists of events thinking of each and every OSV separately is neither easy nor natural way for a mentor to control and think about the virtual model of a baby. Smarter and easier control would be necessary to further improve didactical possibilities of such training.

Physiological Model

We have spent quite some time thinking about the physiological model, which would correctly describe all interdependencies between the listed variables. But after we have had already specified some parts of the complete system, as for example neonatal heart rate mechanism in Fig. 5, we abandoned the idea of physiological modelling due to its tremendous complexity. We decided to simplify the inner mechanisms of the virtual baby by adopting less sophisticated model which takes into account a single control variable.

One-Parameter Description

This simplified model can be reasonably justified from the physiological point of view if we directly relate our governing parameter to the oxygen level in the blood (O_2). In such case the overall complexity of the central nervous system (CNS) is covered by a *black-box* abstraction as seen in Fig. 6. Our task is to specify relations between O_2 Level and observable state variables (HR, RR, Col and Mov).

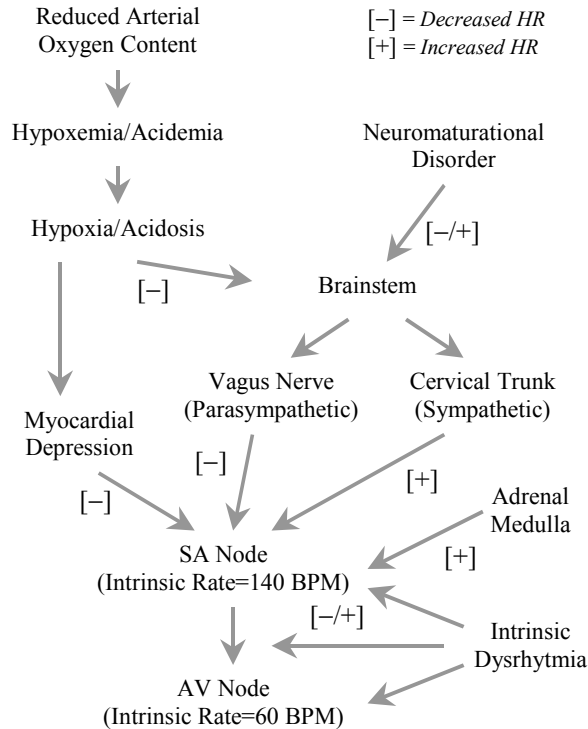


Fig. 5: Neonatal Heart Rate

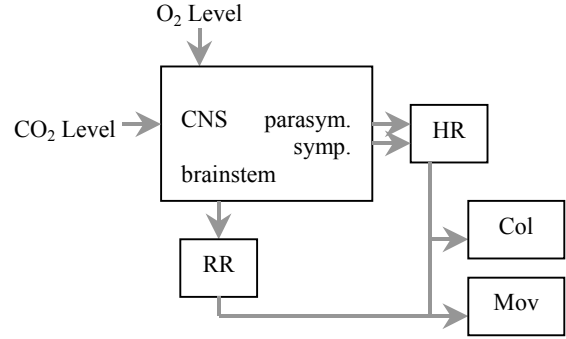


Fig. 6: 'Black-box' relationship between the model governing parameter (O_2 level) and observable state variables (HR, RR, Col, Mov).

Implementation with Partly Linear Relationships

To avoid direct reference to physiology we named the single control variable *Health Index (HI)* in our programming implementation. Its value is in the range $[0,1]$ and the relations $HR(HI)$, $RR(HI)$, $Col(HI)$ and $Mov(HI)$ are described with segments of linear functions - polylines. We call them *OSV Lookup Tables (LTs)* and they can be set arbitrary for different training cases, example is in Fig. 7.

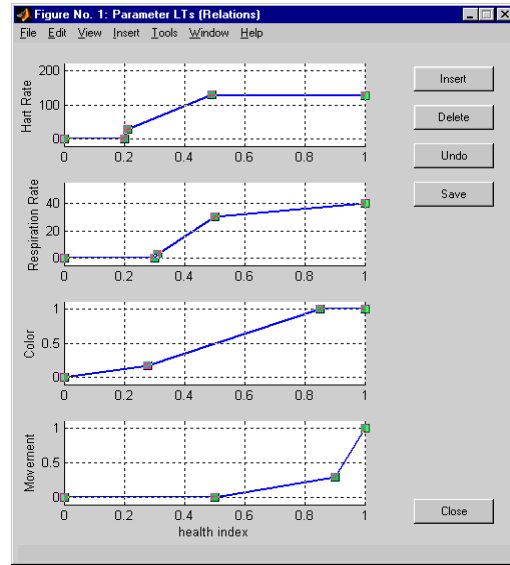


Fig. 7: Partly linear relations between the control variable HI and OSVs.

V. SOFTWARE FOR MODEL TUNING AND VALIDATION

Mentor can create and tune this relations using an independent piece of software - *VIDERO Simulator and Animator*. After setting the relations of OSVs vs. HI in the first window one can interactively specify the trajectory of the HI over time (bottom graph in Fig. 8) and obtain corresponding graphs for each of the four OSVs (upper four graphs in Fig. 8) in the second window. This can be done either in real or simulated (faster) time.

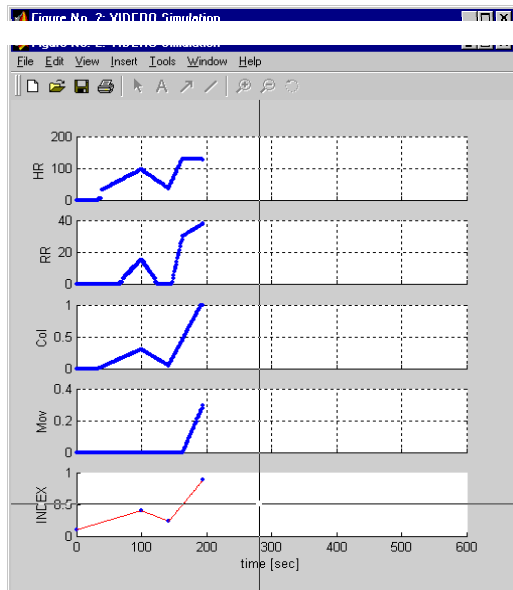


Fig. 8: Graphs of OSVs versus time according to the HI trajectory (bottom graph) and corresponding LTs.

If real time is chosen the animated audio/visual representation of the vital signs can be monitored in the third window during simulation (Fig. 9). In case the trajectories are generated in faster time, simultaneous animation of vital signs can not be generated, but the complete course of animation can be played back afterwards in real time. During the animation numeric values of all parameters are also displayed.

With the described tool a neonatal resuscitation specialist should be able to construct close-to-realistic behaviour of a virtual baby for certain physiological conditions. Defined LTs and HI trajectories can be automatically saved as set-up and scenario files for the target virtual training environment, which will then be used during simulated training.

VI. CONCLUSION

We have described the idea and our prototype system for neonatal resuscitation training using computer-generated multi-user virtual environment built with VRML and Java. Avatar of a new-born baby was made to represent essential vital signs which a student should continuously assess and then properly react according to them. Values of these observable variables can be directly controlled by the mentor, who can also prepare lists (scenarios) of discrete events to define their trajectories in advance.

The major drawback of the current implementation is that each vital sign is treated separately, which obscures the clarity of the virtual baby behaviour and makes control over it difficult. To overcome this we have designed a scheme of partly linear relationships with respect to only one control variable. In physiological terms this single control variable can be approximately related to the oxygen level in the blood. One can, of course, argue that this model is too simplistic

compared to the whole complexity of real physiological processes. This is true, however the advantage of having a possibility to govern four variables by only one control value during the supervised training is way more important than possible slight deviations from the accurate physiological behaviour of a real baby. After all - this behaviour is assessed by the student from the synthetic 3D audio/visual representation where scale of errors during the assessment is by far larger than deviations caused by the model itself. Considering the later fact the benefits of the introduced model are greater than the lost degree of freedom compared to manipulating each parameter separately.

The presented approach is fine for the purpose of our training supervised by a mentor according to the idea in Fig. 2. But in case we would want to create an autonomous tool for unsupervised training this mechanism should be further developed and upgraded with a kind of a medical expert system. Some of our current efforts are aimed to explore this topic also.

ACKNOWLEDGEMENT

This work has been partly supported by the anonymous donation to the Stanford University School of Medicine.

REFERENCES

- [1] D. Korošec, "Building Virtual Environments for Training of Health Care Professionals", Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics, July 23-26, 2000, Orlando, Florida. Vol. 10, pp. 339-342.
- [2] R. S. Bloom, C. Cropley et al., AHA/AAP Neonatal Resuscitation Textbook, 1994, American Heart Association
- [3] Louis P. Halamek, "Development of a Simulated Delivery Room", presented at the 7th Annual Medicine Meets Virtual Reality Conference (MMVR 1999), January 20 - 23, 1999, San Francisco, CA
- [4] R. Carey, G. Bell, The annotated VRML 2.0 reference manual, Addison Wesley, 1997
- [5] http://www.vrml.org/fs_specifications.htm, "Specification for a Standard Humanoid", Web3D Consortium - Humanoid Animation Working Group.

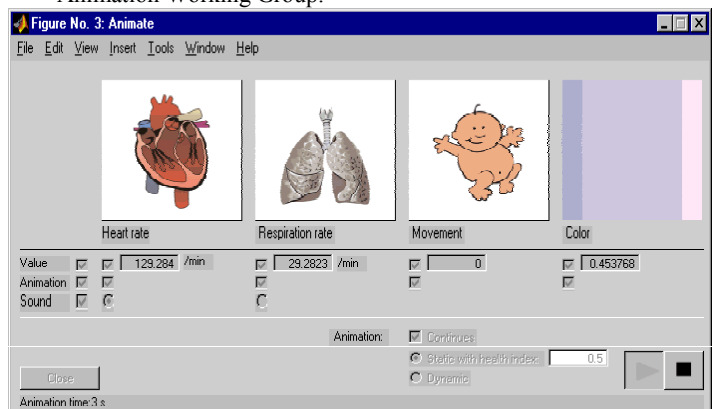


Fig. 9: Animation window of the VIDERO Simulator & Animator program.